RFC 4821 : Packetization Layer Path MTU Discovery

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 4 mai 2007

Date de publication du RFC : Mars 2007

https://www.bortzmeyer.org/4821.html

La détermination de la MTU disponible entre deux nœuds de l'Internet n'a jamais été facile. Ce RFC propose un nouveau protocole pour cette tâche. (Il a depuis été modifié et étendu par le RFC 8899 [1].)

Entre deux machines quelconques connectées à l'Internet, par exemple un serveur HTTP www.example.org et son client, il est très souhaitable de pouvoir déterminer la MTU maximum du chemin qui les relie ("*Path MTU*"). En IPv6, c'est indispensable, les routeurs intermédiaires n'ayant pas le droit de fragmenter les paquets mais c'est également recommandé en IPv4, la fragmentation faisant chuter les performances (l'excellente page <http://www.psc.edu/~mathis/MTU/> décrit en détail pourquoi il serait bon d'augmenter les MTU).

La méthode standard est décrite dans le RFC 1191 (et RFC 1981 pour IPv6) et est connue sous le nom de "*Path MTU discovery*". Elle consiste à envoyer des paquets avec le bit DF ("*Don't Fragment*") mis et à attendre les messages ICMP "*Packet Too Big*" (le vrai nom de ce message, spécifié dans le RFC 792 est "*fragmentation needed and DF set*"). Mise en œuvre depuis longtemps dans tous les systèmes, elle peut se tester avec certaines versions de traceroute qui disposent de l'option -M :

```
% traceroute-nanog -M 192.0.2.42
traceroute to 192.0.2.42, 64 hops max
MTU=1500
...
MTU=1492
```

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc8899.txt

On voit le passage de la MTU de départ du client (les 1500 octets d'Ethernet) à celle du chemin complet, limité à 1492 octets, probablement par une encapsulation PPPoE. (On peut aussi utiliser un tel outil depuis le Web en <http://www.ncne.org/jumbogram/mtu_discovery.php>.)

Mais cette méthode a un défaut : il faut que les paquets ICMP arrivent. Or, beaucoup de sites filtrent stupidement tout ICMP sur leur coupe-feu et, en pratique, cette méthode n'est donc pas fiable (le RFC 2923 détaille pourquoi).

Notre RFC propose donc une alternative, ne dépendant pas de la réception des paquets ICMP et fonctionnant donc en présence de « trous noirs » qui absorbent tous ces paquets ICMP. Il suggère tout simplement de tenir compte des paquets perdus, en supposant que si seuls les plus gros se perdent, c'est probablement qu'ils étaient plus gros que la MTU. La nouvelle méthode est donc d'essayer des paquets de différentes tailles et de surveiller les pertes. Les détails de l'implémentation dépendent du protocole utilisé. La nouvelle méthode se nomme PLPMTUD ("*Packetization Layer Path MTU Discovery*").

Les protocoles comme TCP surveillant déjà les pertes de paquets, la modification nécessaire serait donc raisonnable (section 10.1). Notre RFC décrit aussi comment réaliser cette recherche de MTU pour d'autres protocoles comme SCTP ou même au niveau applicatif (section 10.4).

Notre RFC détaille aussi les pièges possibles. Par exemple, si certains équipements réseaux ont un comportement non-reproductible (la section 4 cite le cas de répéteurs Ethernet qui ne refusent pas les paquets trop gros mais n'arrivent pas non plus à les transmettre de manière fiable, leur horloge n'étant pas stable sur une période suffisamment longue), ce protocole ne peut pas fonctionner.

Voilà, mais rappelez-vous que la procédure décrite ici a depuis été mise à jour dans le RFC 8899 notamment pour les autres protocoles que TCP.